FIUBA-PROG

Manual del Usuario

66.09 Laboratorio de Microcomputadoras
Pedro Ignacio Martos
pmartos@fi.uba.ar / pimartos@gmail.com

INDICE

Descripcion del Sistema 3
Conexionado para la programacion de microcontroladores AT89Sxx 3
Descripcion del Software 3
Archivo de configuracion 4
Comandos Disponibles 4
Comando ERASE 4
Comando RESET 4
Comando WRITEFILE 4
Comando WRITEFUSES 5
Comando READFUSES 5
Comando SIGNATURE 5
Comando VERSION 5
Secuencia de comandos para la programacion 6

Por preguntas, comentarios o sugerencias de mejoras, se puede contactar al autor en los correos electronicos antes mencionados.

Descripcion del Sistema

El programador “Fiuba-prog” es un dispositivo basado en el circuito integrado FT2232 de la empresa Future Technologies. El diseño del circuito impreso esta basado en la placa OOCDLINK, por lo que tambien tiene la funcionalidad de trabajar como interfase JTAG a traves del software OpenOCD. Asimismo se ha agregado un buffer 74LVC245, lo que permite implementar la interfase C2 y programar los microcontroladores de la empresa SiLabs, tales como el C8051F320.
Este programador se conecta a los dispositivos AT89Sxx a través de su interfase para In-System Programming (ISP); es decir, utiliza las señales de programacion MISO, MOSI y SCK. Actualmente se encuentra implementada la capacidad de programar los siguientes dispositivos:

- AT89S52
- AT89S2051
- AT89S8253

Para ello se utiliza la herramienta de software por linea de comanto ft89prog; la cual permite la programación de los dispositivos antes mencionados.

Conexionado para la programacion de microcontroladores AT89Sxx

El conexionado para realizar la programacion de los microcontroladores es el siguiente: Mediante un jumper se mantiene la entrada de RESET del microcontrolador en el estado lógico ‘1’ (microcontrolador en reset); y se conectan las señales de ISP según la siguiente tabla:

HD04 (Programador)	ISP (AT89Sxx)
1: GND	20: GND
3: VREF	40: VCC
11: D3	8: P1.7 / SCK
15: D5	6: P1.5 / MOSI
17: D6	7: P1.6 / MISO

Asimismo se recomienda ver el circuito esquematico del programador.

Descripcion del Software

La herramienta de software a utilizar es la version para Windows del programa FT89PROG. El codigo fuente del mismo se encuentra disponible en http://sourceforge.net/projects/ft89prog/
El ejecutable disponible (ft89prog.exe) es para la plataforma Windows XP de 32 bits.

Se debe verificar que se encuentren instalados:

- El driver para el FT2232 (CDM20814_Setup.exe). El mismo se puede obtener en http://www.ftdichip.com/Drivers/D2XX.htm
- Las librerias Run-Time de 32 bits de Visual Studio 2008 SP1 (vcredist_x86). Las mismas se pueden obtener en http://www.microsoft.com/downloads/es-es/details.aspx?familyid=a5c84275-3b97-4ab7-a40d-3802b2af5fc2&displaylang=es

Archivo de configuracion

El archivo de configuracion (ft89progrc) es un archivo de texto en formato XML que indica cuales pines del circuito integrado FT2232 estan asignados a las señales ISP. Se recomienda no alterar esta configuracion. Asimismo este archivo indica cual es el dispositivo a programar (por defecto esta configurado el AT89S52); debe cambiarse al dispositivo que se esta utilizando.

Comandos Disponibles

Comando ERASE

Este comando realiza el borrado total de la memoria de programacion del microcontrolador.

Comando RESET

Este comando fuerza al estado lógico '1' al pin definido como "RST" en el archivo de configuracion. Dada la capacidad de corriente del circuito integrado FT2232, es recomendable que inicialmente esta opcion no se utilice y se fije el estado del pin de reset del microcontrolador en forma externa mediante un jumper.

Comando WRITEFILE

Este comando escribe un archivo en formato binario (.bin) al microcontrolador. El mismo debe tener previamente su memoria flash borrada.
Dado que las herramientas de programacion generan normalmente un archivo en formato intel HEX (.hex); se utiliza la herramienta HEX2BIN (disponible en el sitio web de Keil, http://www.keil.com) para pasar del formato .hex al formato .bin.

Comando WRITEFUSES

Este comando fija el valor de los fusibles de seguridad de los microcontroladores AT89Sxx. Se recomienda inicialmente no utilizar esta funcion.

Comando READFUSES

Este comando lee el estado de los fusibles de seguridad del microcontrolador conectado al programador.

Comando SIGNATURE

Este comando lee la firma digital del microcontrolador conectado y verifica si es el mismo que esta en el archivo de configuracion

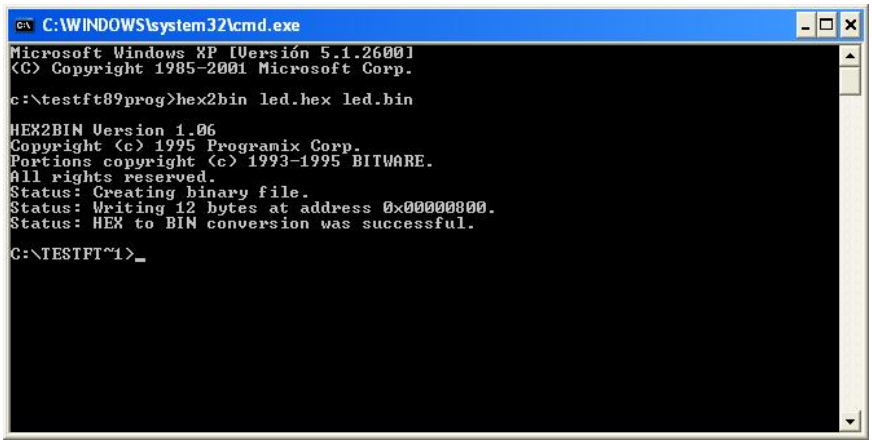
Comando VERSION

Este comando muestra la version actual del software

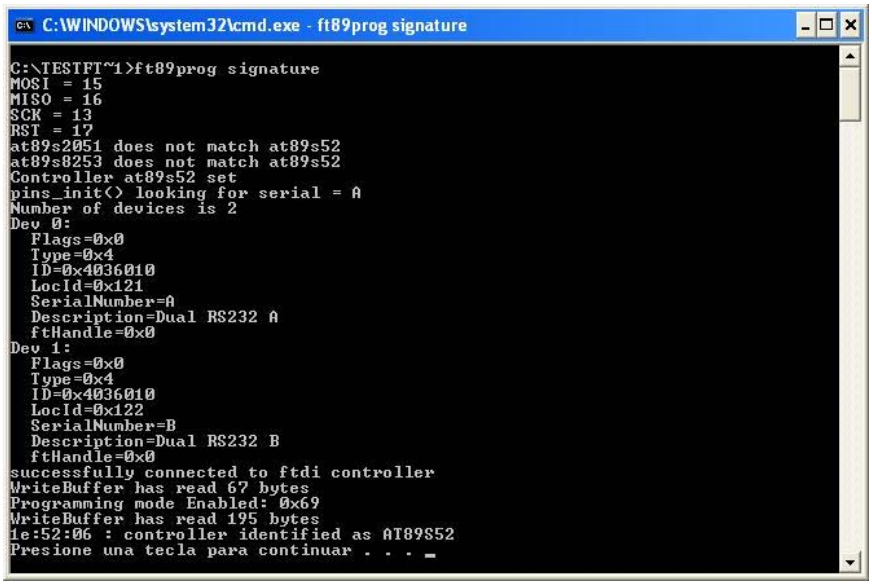
Secuencia de comandos para la programacion

Para la programacion de un microcontrolador se debe realizar la siguiente secuencia de pasos:

1. Una vez simulado exitosamente el programa, se genera la imagen de memoria en formato .HEX
2. Con la herramienta Hex2bin se convierte el archivo .hex al formato .bin

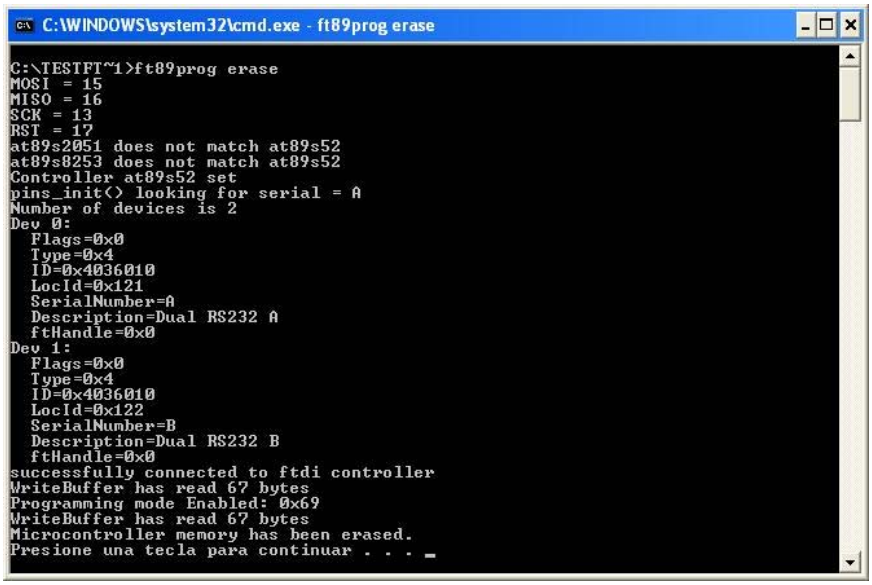

C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
c:\testft89prog>hex2bin led.hex led.bin
HEX2BIN Version 1.06
Copyright (c) 1995 Programix Corp.
Portions copyright (c) 1993-1995 BITWARE.
All rights reserved.
Status: Creating binary file.
Status: Writing 12 bytes at address 0x00000800.
Status: HEX to BIN conversion was successful.
C:\TESTFT~1>

3. Se conecta el microcontrolador al programador
4. Se fija el pin de reset del microcontrolador al estado logico ‘1’
5. Se verifica la conexión entre el programador y el microcontrolador con el comando SIGNATURE



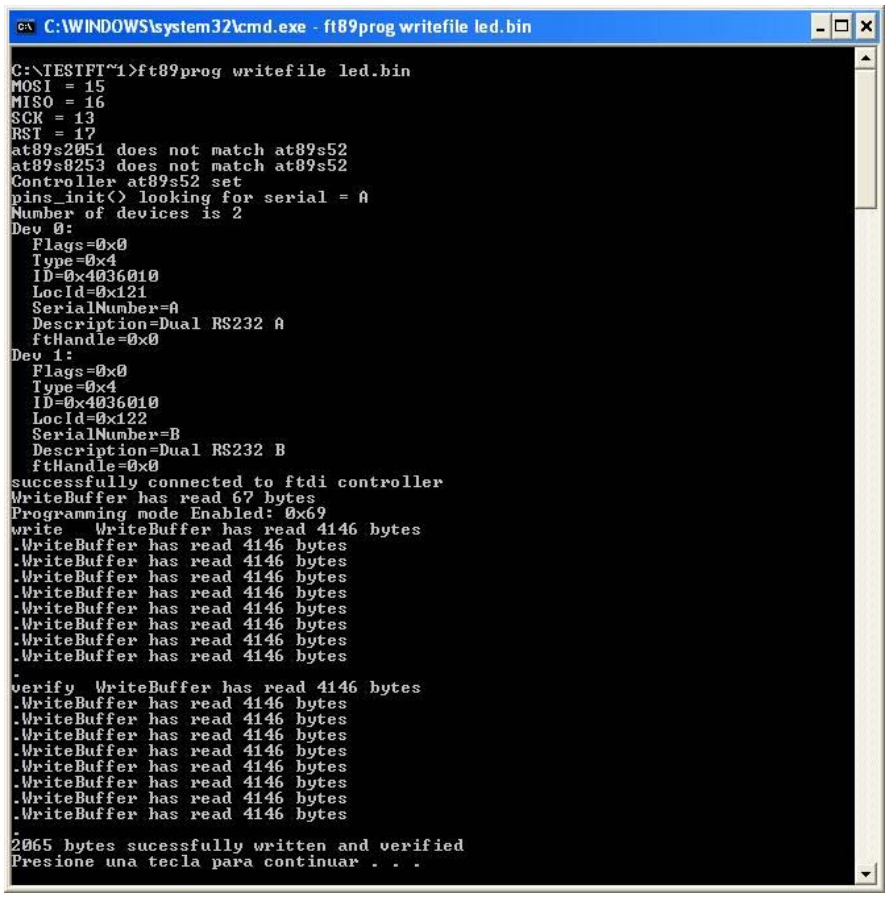
Verificar que se haya entrado exitosamente al modo de programacion: Programming Mode Enabled: 0x69

6. Se borra el contenido de la memoria del microcontrolador con el comando ERASE



Verificar que se haya entrado exitosamente al modo de programacion: Programming Mode Enabled: 0x69

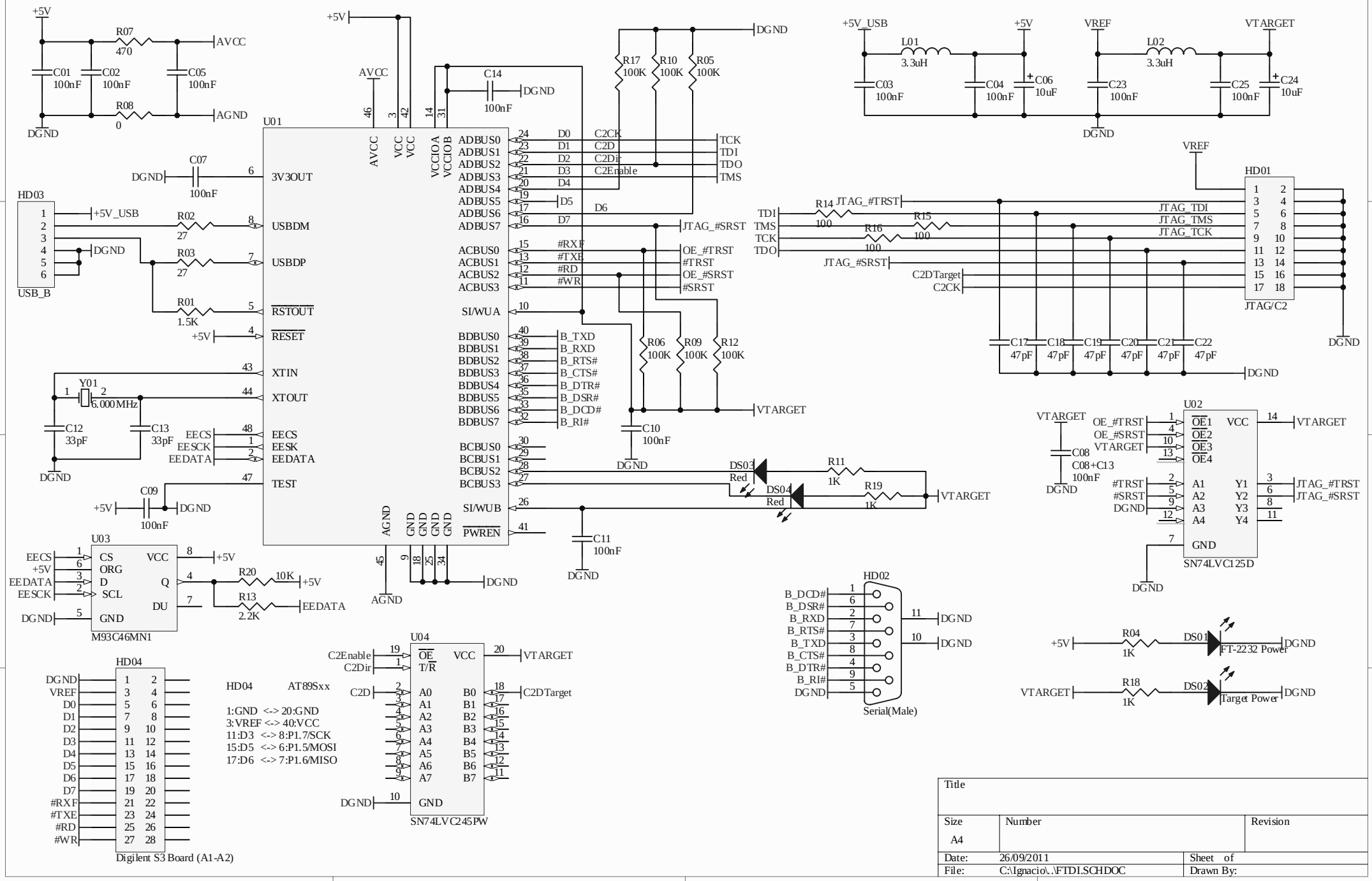
7. Se programa el microcontrolador con el comando WRITEFILE



Verificar que se haya entrado exitosamente al modo de programacion: Programming Mode Enabled: 0x69

8. Se fija el pin de reset del microcontrolador en el estado lógico ‘0’ y se verifica el funcionamiento del programa.

NOTA: Debido a las particularidades del puerto USB de algunas computadoras (host OHCI vs host UHCI), puede ser necesario desconectar y reconectar el programador al puerto USB. Es por ello que en la ejecucion de cada comando se debe verificar que se haya entrado exitosamente al modo de programacion.


+5V
R07
470
AVCC
C01
100nF
C02
100nF
C05
100nF
R08
0
AGND
DGND
U01
AVCC
VCC
VCCIOA
VCCIOB
C14
100nF
R17
100K
R10
100K
R05
100K
ADBUS0
ADBUS1
ADBUS2
ADBUS3
ADBUS4
ADBUS5
ADBUS6
ADBUS7
D0
D1
D2
D3
D4
D5
D6
D7
C2CK
C2D
C2Dir
C2Enable
TCK
TDI
TDO
TMS
JTAG_#SRST
C07
100nF
3V3OUT
HD03
USB_B
+5V_USB
R02
27
USBDM
DGND
R03
27
USBDP
R01
1.5K
RSTOUT
RESET
ACBUS0
ACBUS1
ACBUS2
ACBUS3
#RXF
#TXE
#RD
#WR
OE_#TRST
#TRST
OE_#SRST
#SRST
SI/WUA
BDBUS0
BDBUS1
BDBUS2
BDBUS3
BDBUS4
BDBUS5
BDBUS6
BDBUS7
B_TXD
B_RXD
B_RTS#
B_CTS#
B_DTR#
B_DSR#
B_DCD#
B_RI#
R06
100K
R09
100K
R12
100K
VTARGET
C10
100nF
XTIN
XTOUT
Y01
6.000MHz
C12
33pF
C13
33pF
EECS
EESCK
EEDATA
TEST
C09
100nF
BCBUS0
BCBUS1
BCBUS2
BCBUS3
SI/WUB
PWREN
C11
100nF
DS03
Red
DS04
Red
R11
1K
R19
1K
AGND
GND
U03
CS
ORG
D
SCL
VCC
Q
DU
GND
M93C46MN1
R20
10K
R13
2.2K
HD04
AT89Sxx
1:GND <-> 20:GND
3:VREF <-> 40:VCC
11:D3 <-> 8:P1.7/SCK
15:D5 <-> 6:P1.5/MOSI
17:D6 <-> 7:P1.6/MISO
Digilent S3 Board (A1-A2)
U04
C2Enable
C2Dir
C2D
OE
T/R
A0
A1
A2
A3
A4
A5
A6
A7
B0
B1
B2
B3
B4
B5
B6
B7
C2DTarget
SN74LVC245PW
HD02
B_DCD#
B_DSR#
B_RXD
B_RTS#
B_TXD
B_CTS#
B_DTR#
B_RI#
Serial(Male)
+5V_USB
L01
3.3uH
C03
100nF
C04
100nF
C06
10uF
VREF
L02
3.3uH
C23
100nF
C25
100nF
C24
10uF
VTARGET
HD01
JTAG_TDI
JTAG_TMS
JTAG_TCK
JTAG_#TRST
JTAG_#SRST
C2DTarget
C2CK
R14
100
R15
100
R16
100
JTAG/C2
C17
47pF
C18
47pF
C19
47pF
C20
47pF
C21
47pF
C22
47pF
U02
OE1
OE2
OE3
OE4
VCC
A1
A2
A3
A4
Y1
Y2
Y3
Y4
GND
SN74LVC125D
C08
C08+C13
100nF
R04
1K
DS01
FT-2232 Power
R18
1K
DS02
Target Power
Title
Size
A4
Number
Revision
Date:
26/09/2011
Sheet of
File:
C:\Ignacio\..\FTDI.SCHDOC
Drawn By: